

# Sistema para Otimização baseada em Meta-Heurística de Redes Neurais por Convolução

Universidade Estadual Paulista “Júlio de Mesquita Filho”  
Faculdade de Ciências  
Departamento de Computação

Trabalho de Conclusão de Curso

Bárbara Caroline Benato  
Orientador: Prof. Dr. Aparecido Nilceu Marana

# Introdução

- Técnicas de aprendizado de máquina utilizadas para a extração de características (sinais ou imagens).
- Conjunto de treinamento como base -> aprendizado da técnica -> classificação conjunto de dados (DUDA; HART; STORK, 2000).
- Otimização: estudo e seleção dos melhores parâmetros responsáveis por conduzir uma melhor taxa de reconhecimento.

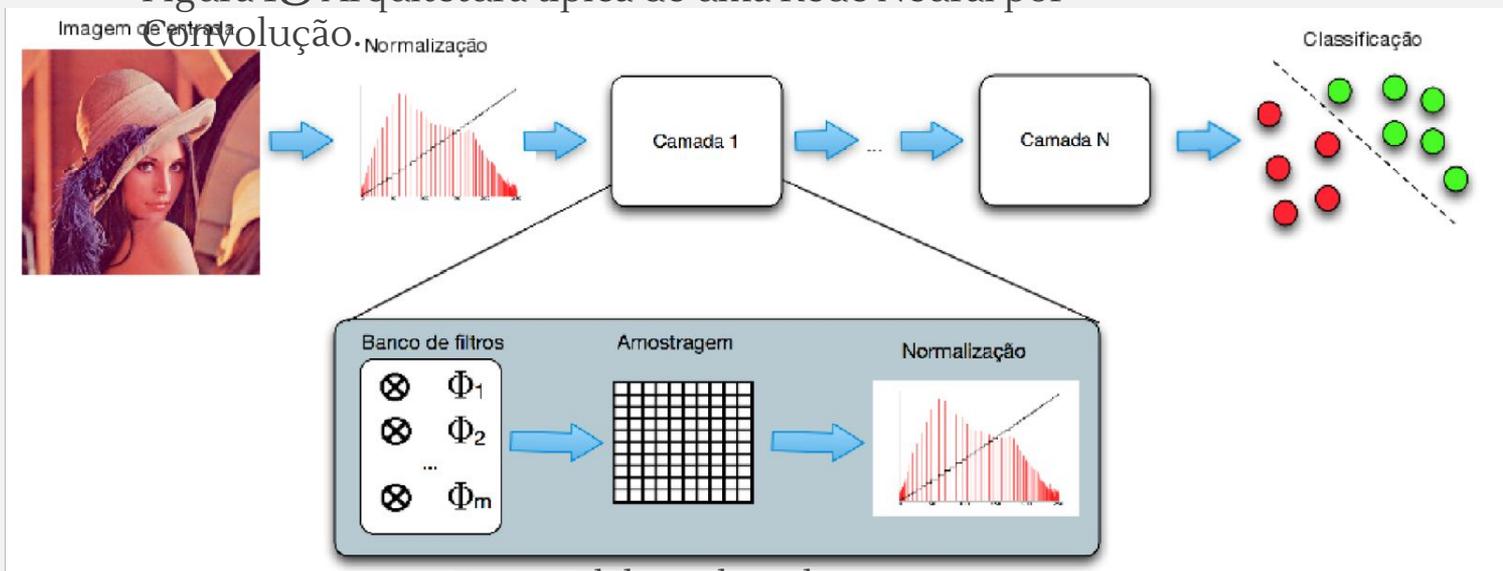
# Introdução

- Técnicas de aprendizado de máquina: Aprendizado em Profundidade (*Deep Learning*).
- Caracterizada por hierarquia no momento de extração e aprendizado do conhecimento.
- Baseadas no processamento de informações do cérebro humano (hierárquica): cada etapa é responsável por extrair e codificar diferentes níveis de informação.

# Introdução

Redes Neurais por Convolução (*Convolutional Neural Networks - CNNs*) (LECUN et al., 1998) diferentes estágios: filtragem, normalização e sub-amostragem.

Figura 1 - Arquitetura típica de uma Rede Neural por



Fonte: Elaborado pelo autor.

# Introdução

Grande quantidade de parâmetros:  
- número e tamanho de cada filtro, nível de sub-amostragem de uma imagem.

Parâmetros ainda podem ser distintos entre cada etapa (camada) de execução de uma CNN.

# Introdução

Buscas aleatórias para “popular” o conjunto de parâmetros de uma CNN.

Técnicas de otimização por meta-heurísticas têm obtido resultados mais eficazes do que uma busca aleatória (FEDOROVICI et al., 2012).

Otimização das Aves Migratórias (*Migration Birds Optimization - MBO*) (DUMAN; UY- SAL; ALKAYA, 2012): baseado no comportamento de vôo de aves em bando, reduzem o gasto de energia e aumentam o alcance de vôo.

# Introdução

Figura 2 – Diferença entre os fluxos de ar nas partes superior e inferior da asa de uma ave.



Fonte: Elaborado pelo autor.

# Objetivos

Desenvolvimento de sistema capaz de:

- validar o uso de meta-heurísticas para otimização de técnicas de aprendizado de máquina, Redes Neurais por Convolução.
- facilitar e simplificar a tarefa de otimizar Redes Neurais por Convolução para usuários das bibliotecas Caffé e LibOPT.

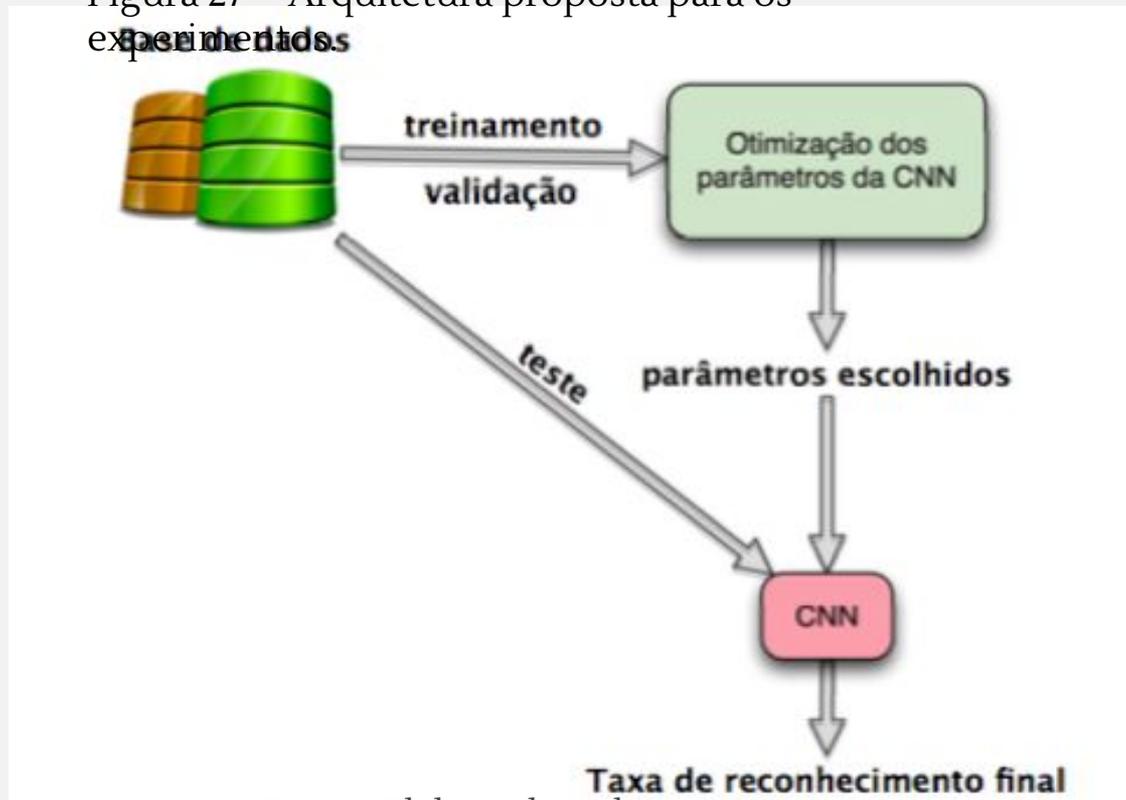
# Motivação

- Elevado número de parâmetros da técnica de aprendizado em profundidade.
- Metodologia empregada no processo de otimização de Redes Neurais por Convolução.

# Especificação do Sistema

- Metodologia

Figura 27 – Arquitetura proposta para os experimentos

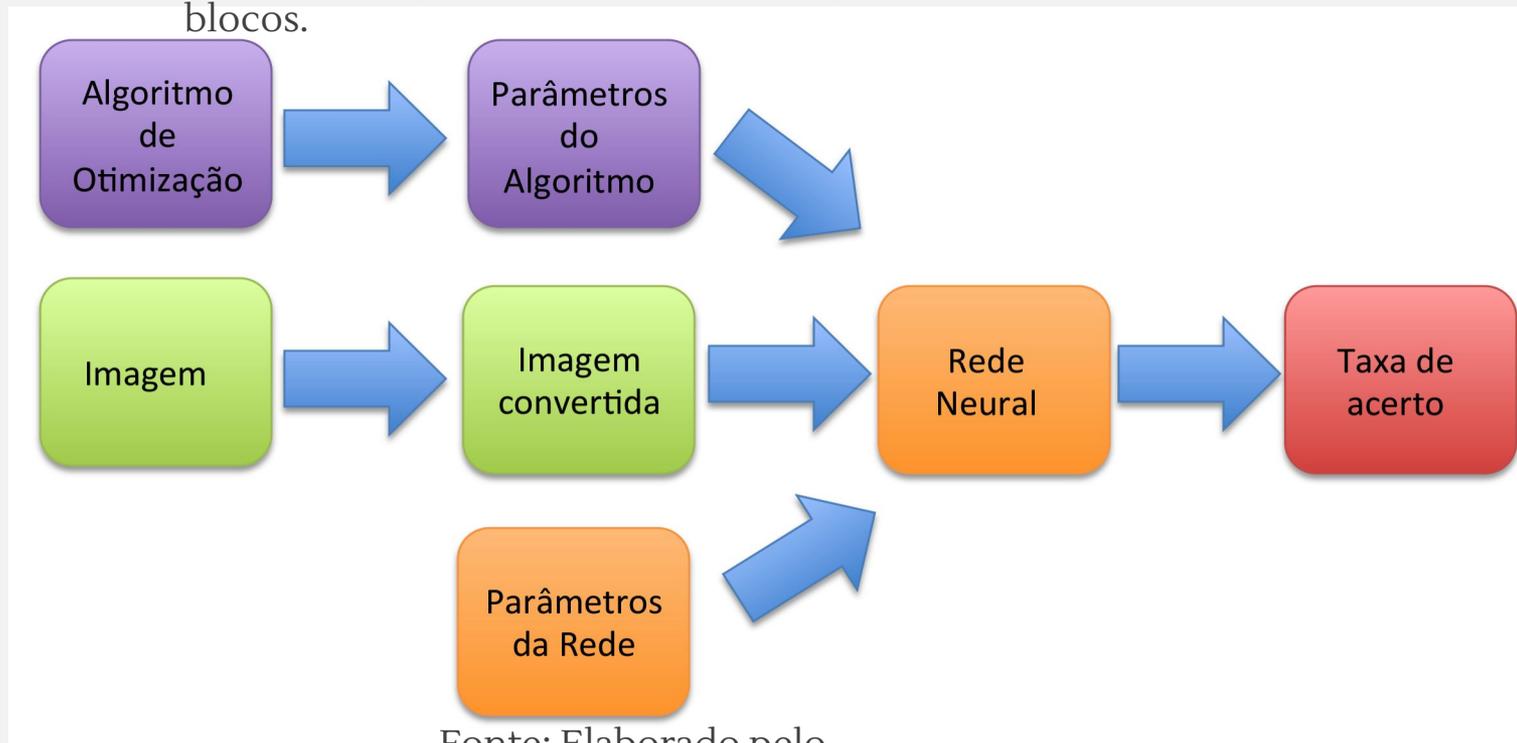


Fonte: Elaborado pelo autor.

# Especificação do Sistema

- Módulos do Sistema

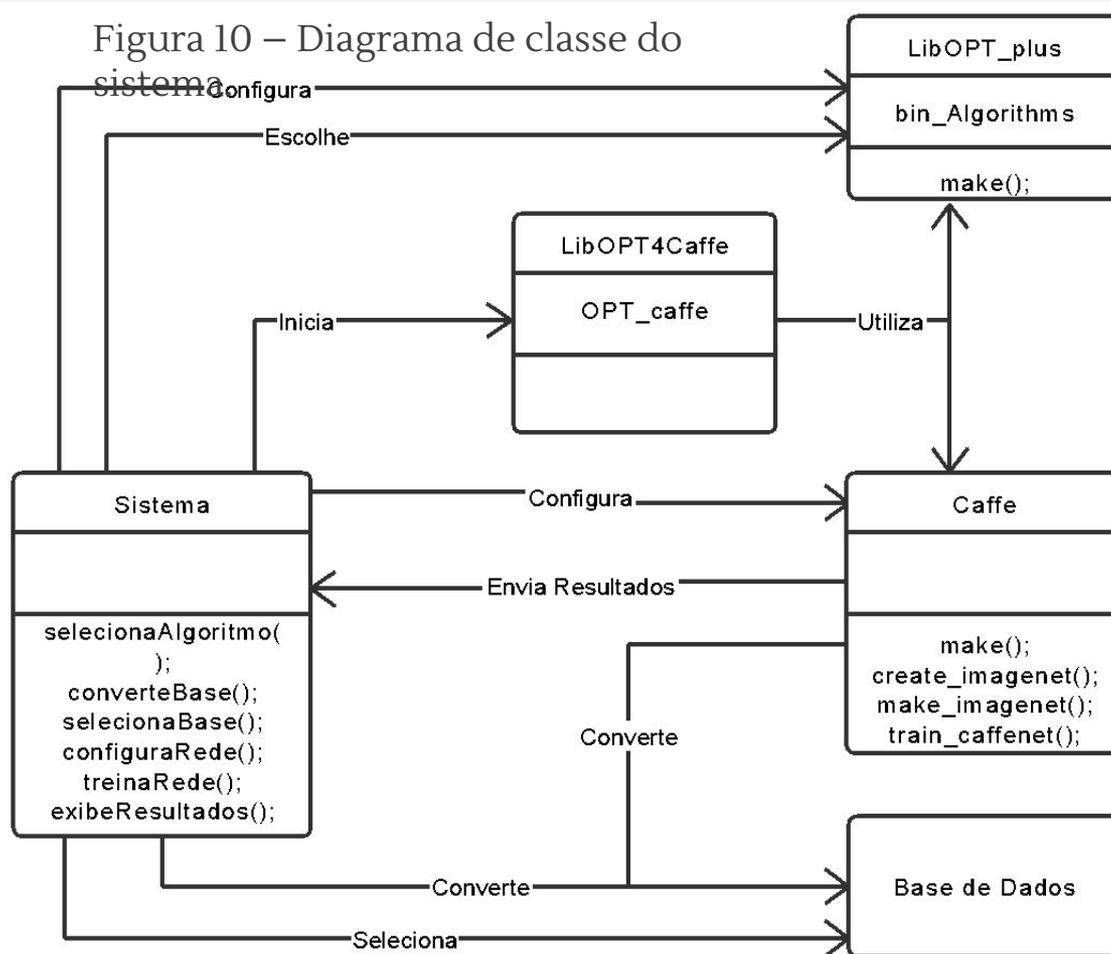
Figura 9 – Representação do sistema por meio de diagrama de blocos.



Fonte: Elaborado pelo autor.

# Especificação do Sistema

- Diagrama de Classes



# Sistema para a Otimização de Redes Neurais por Convolução

CNN: biblioteca de código Caffe (JIA et al., 2014), implementação de código orientado às Unidades de Processamento Gráfico (*GPU*)  
Arquiteturas: LeNet, CIFAR10, ImageNet.

MBO: biblioteca de otimização LibOPT-plus, linguagem de programação C, grupo Recogna.  
Algoritmos: MBO, HS, PSO.

Integração: biblioteca de LibOPT4Caffe (ROSA, 2016).

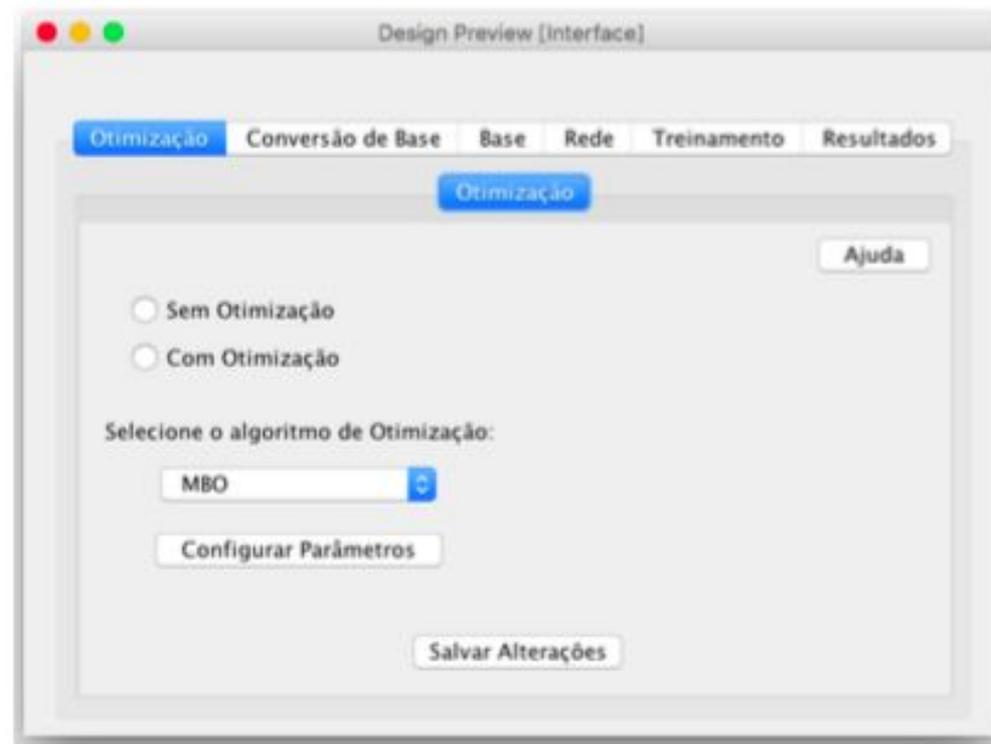
# Sistema para a Otimização de Redes Neurais por Convolução

Sistema: desenvolvido em linguagem Java, pacote JSch, integração com os comandos da linguagem ShellScript do sistemas baseados no modelo UNIX (OPENGROUP, 1995).

Ambiente de desenvolvimento: NetBeans IDE (ORACLE, 2016).

# Sistema para a Otimização de Redes Neurais por Convolução

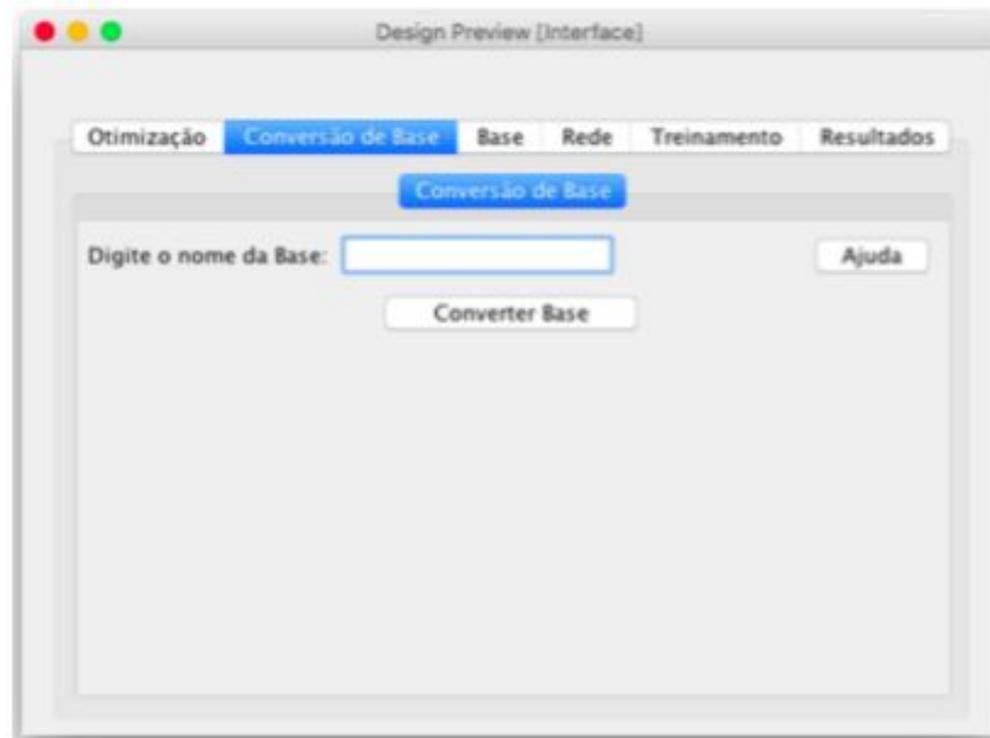
Figura 17 – Aba Otimização.



Fonte: Elaborado pelo autor.

# Sistema para a Otimização de Redes Neurais por Convolução

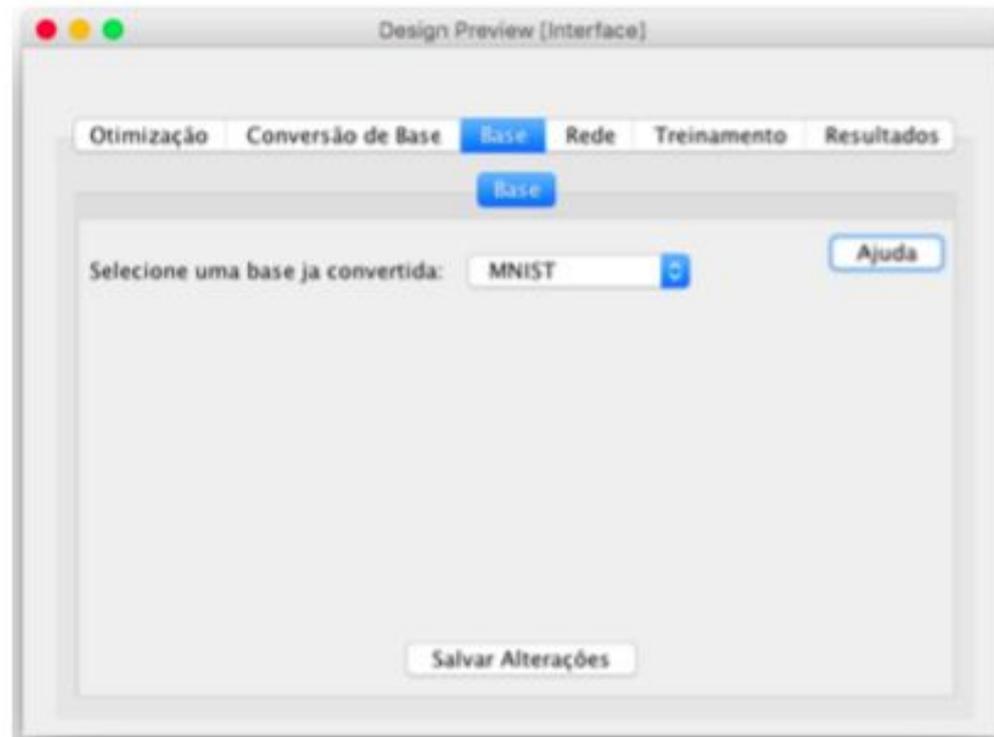
Figura 18 – Aba Conversão de Base.



Fonte: Elaborado pelo autor.

# Sistema para a Otimização de Redes Neurais por Convolução

Figura 19 – Aba Base.



Fonte: Elaborado pelo autor.

# Sistema para a Otimização de Redes Neurais por Convolução

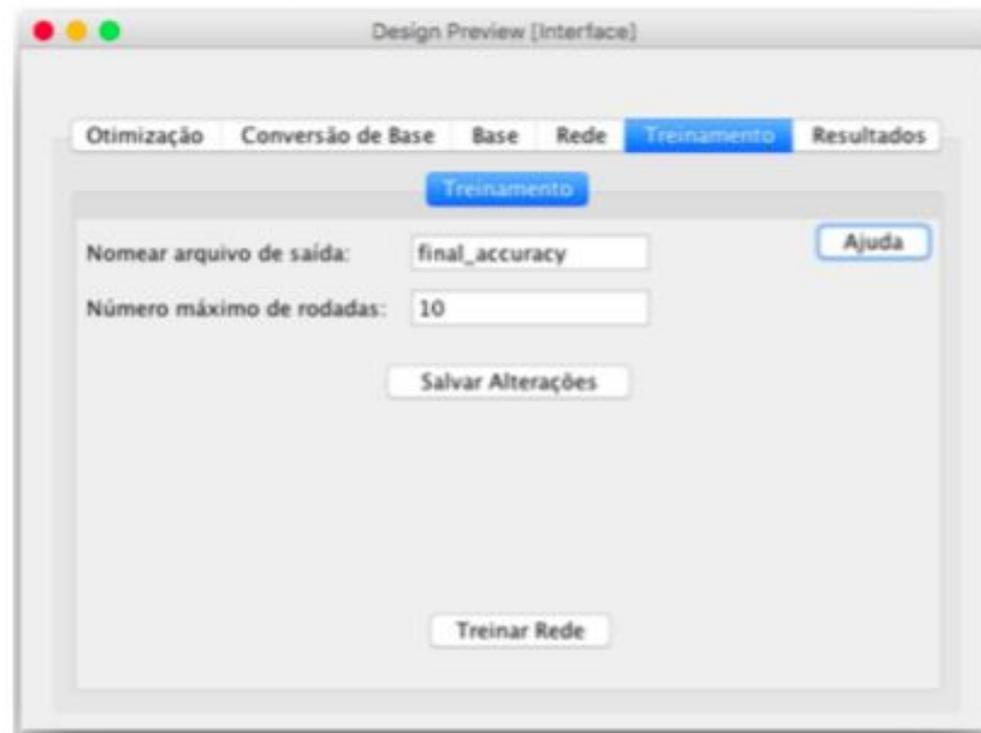
Figura 20 – Aba Rede.



Fonte: Elaborado pelo autor.

# Sistema para a Otimização de Redes Neurais por Convolução

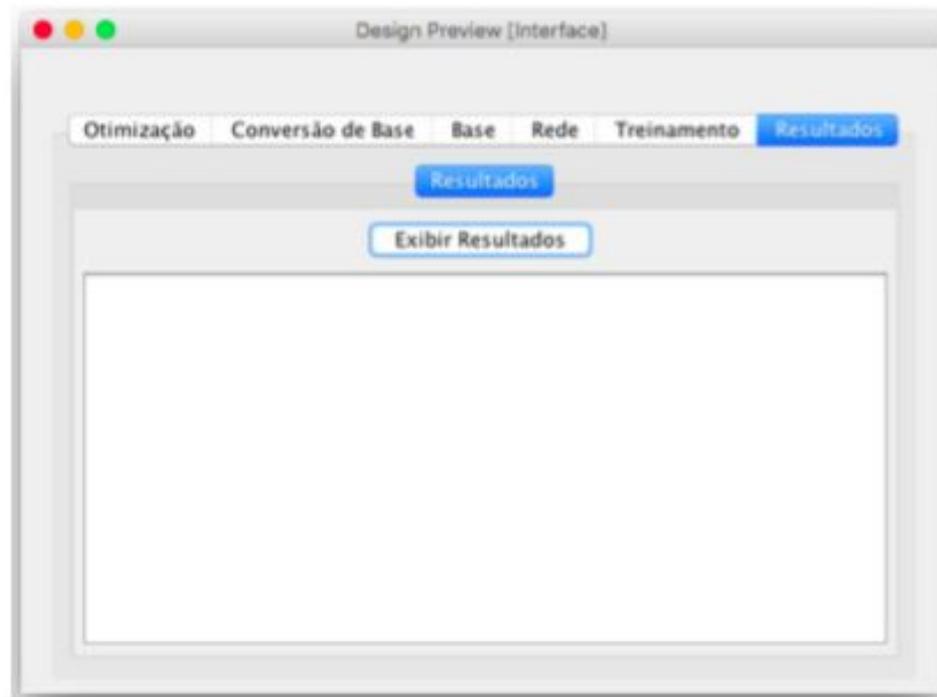
Figura 21 – Aba Treinamento.



Fonte: Elaborado pelo autor.

# Sistema para a Otimização de Redes Neurais por Convolução

Figura 22 – Aba Resultados.



Fonte: Elaborado pelo autor.

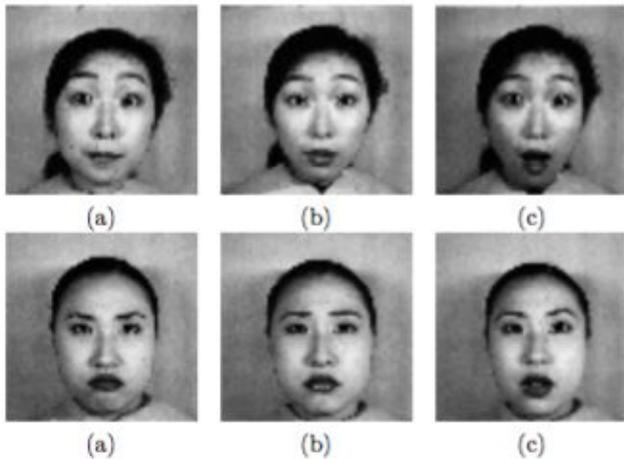
# Utilização do Sistema para o Reconhecimento de Expressões Faciais

Função objetivo: função guiar os agentes evolucionistas busca pela melhor solução global. Presente problema: taxa de acerto de reconhecimento da expressão facial sobre um conjunto de validação.

Base de dados: treinamento, validação e teste. Dois primeiros utilizados para o aprendizado dos parâmetro, teste utilizado apenas para a verificação da taxa de reconhecimento final.

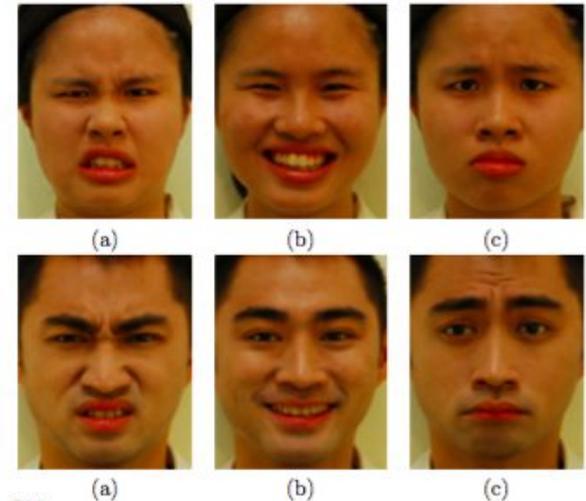
# Utilização do Sistema para o Reconhecimento de Expressões Faciais

Figura 6 – Exemplos de diferentes expressões faciais da base JAFFE.



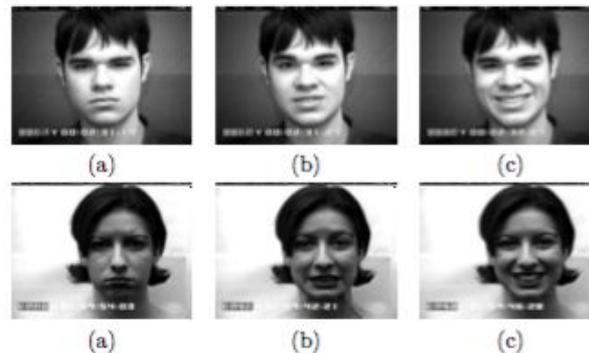
Fonte: Imagens da base JAFFE<sup>5</sup>.

Figura 8 – Exemplos de diferentes expressões faciais da base TFEID.



Fonte: Imagens da base TFEID<sup>7</sup>.

Figura 7 – Exemplos de diferentes expressões faciais da base CK.



Fonte: Imagens da base CK<sup>6</sup>.

# Utilização do Sistema para o Reconhecimento de Expressões Faciais

Cada agente executa uma CNN para extração de características das imagens dos conjuntos de treinamento e validação. Após o processo de convergência, obtem-se melhor conjunto de parâmetros.

# Utilização do Sistema para o Reconhecimento de Expressões Faciais

Para fins de comparação, foi utilizado o algoritmo Busca Harmônica (Harmony Search - HS) (GEEM, 2009).

Foi realizada a validação cruzada com 10 rodadas.

Para as técnicas de otimização, 15 soluções no espaço de busca foram consideradas (aves e harmonias) em 50 iterações para o HS e 25 iterações para o MBO como critério de convergência.

Tabela 1 – Configuração dos parâmetros.

Técnica	Parâmetros
MBO	$k = 5, X = 2, M = 2$
HS	$HMCR = 0.7, PAR = 0.5, \rho = 0.1$

# Resultados Obtidos

Tabela 2 – Resultados experimentais sobre a base de dados JAFFE.

Técnica	Acurácia Final (conjunto de teste)	#chamadas
Caffe	29,29% $\pm$ 0.03	1
<b>MBO</b>	<b>34,05% <math>\pm</math> 0.09</b>	<b>65</b>
HS	24,74% $\pm$ 0.04	65

Tabela 4 – Resultados experimentais obtidos sobre a base de dados Cohn-Kanade AU-Coded Expression.

Técnica	Acurácia Final (conjunto de teste)	#chamadas
Caffe	67,18% $\pm$ 0.06	1
MBO	65,35% $\pm$ 0.05	65
<b>HS</b>	<b>68,51% <math>\pm</math> 0.05</b>	<b>65</b>

Tabela 6 – Resultados experimentais obtidos sobre a base de dados Taiwanese Facial Expression.

Técnica	Acurácia Final (conjunto de teste)	#chamadas
Caffe	91,17% $\pm$ 0.01	1
<b>MBO</b>	<b>93,40% <math>\pm</math> 0.02</b>	<b>65</b>
HS	92,11% $\pm$ 0.01	65

# Conclusões

Resultados experimentais obtidos confirmam a proposta deste trabalho:

- MBO é passível de ser empregado para a otimização dos parâmetros de uma CNN.
- sistema capaz de simplificar e abstrair um processo trabalhoso de configuração e integração manual das várias bibliotecas (algoritmos/bases).

Presente trabalho de conclusão de curso contribuiu para os estudos a cerca da técnica de aprendizado de máquina aqui explorada.

# Trabalhos Futuros

Interface visual do sistema aperfeiçoada:

- adicionar novas técnicas de otimização
- armazenar bases convertidas em banco de dados (atualização).

Facilitar ainda mais a compreensão dos processos relacionados à otimização e rede neural utilizada: usuário maior aproveitamento do sistema e uso regular.

# Referências

- B., L. P.; A., S. C. Formation flight of birds. *Science*, v. 3934, n. 168, p. 1003–1005, 1970.
- COX, D.; PINTO, N. Beyond simple features: A large-scale feature search approach to unconstrained face recognition. In: *Proceedings of the IEEE International Conference on Automatic Face Gesture Recognition and Workshops*. [S.l.: s.n.], 2011. p. 8–15.
- DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification (2nd Edition)*. [S.l.]: Wiley-Interscience, 2000. ISBN 0471056693.
- DUMAN, E.; UYSAL, M.; ALKAYA, A. Migrating birds optimization: A new metaheuristic approach and its performance on quadratic assignment problem. *Information Sciences*, Elsevier Science Inc., New York, NY, USA, v. 217, p. 65–77, 2012.
- FEDOROVICI, L. et al. Embedding gravitational search algorithms in convolutional neural networks for OCR applications. In: *7th IEEE SACL*. [S.l.: s.n.], 2012. p. 125–130.
- FUKUSHIMA, K.; MIYAKE, S. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognition*, v. 15, n. 6, p. 455–469, 1982.
- GEEM, Z. W. *Music-Inspired Harmony Search Algorithm: Theory and Applications*. 1st. ed. [S.l.]: Springer Publishing Company, Incorporated, 2009. ISBN 364200184X, 9783642001840.
- GEISLER, W.; ALBRECHT, D. Cortical neurons: isolation of contrast gain control. *Vision Research*, v. 32, n. 8, p. 1409–1410, 1992.
- HINTON, G. Training products of experts by minimizing contrastive divergence. *Neural Computation*, MIT Press, Cambridge, MA, USA, v. 14, n. 8, p. 1771–1800, 2002. ISSN 0899-7667.
- JIA, Y. et al. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.
- LECUN, Y. et al. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, MIT Press, Cambridge, MA, USA, v. 1, n. 4, p. 541–551, 1989.
- LECUN, Y. et al. *Advances in neural information processing systems 2*. In: TOURETZKY, D. S. (Ed.). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990. cap. *Handwritten Digit Recognition with a Back-propagation Network*, p. 396–404.
- LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998.
- LTD, B. D. I. T. C. *Processon*. [S.l.]: Processon.com, 2014. <<https://www.processon.com>>.
- MARANA, A.; PAPA, J. *Recogna*. [S.l.]: FC.Unesp, 2010. <<http://www2.fc.unesp.br/papa/recogna/index.html>>.
- OPENGROUP, T. *Unix*. [S.l.]: opengroup.org, 1995. <<http://opengroup.org/unix>>. ORACLE. *NetBeans IDE*. [S.l.]: NetBeans.org, 2016. <[https://netbeans.org/index pt BR. html](https://netbeans.org/index.pt BR. html)>.
- PAPA, J. *LibOPT-plus*. [S.l.]: GitHub, 2016. <<https://github.com/jppbsi/LibOPT-plus>>.
- ROSA, G. *LibOPT4Caffe*. [S.l.]: GitHub, 2016. <<https://github.com/gugarosa/LibOPT4Caffe>>.
- ROSARIO, S. *Facial Emoticons: Reprodu de Informa Associada a Expresses Faciais por Via do seu Reconhecimento*. [S.l.]: Universidade Tica de Lisboa, 2018. <<https://fenix.tecnico.ulisboa.pt/cursos/meic-a/dissertacao/2353642189428>>.
- TEIXEIRA, A. *Desenvolvimento de uma Interface Grca para Classificadores de Imagem*. [S.l.]: DocPlayer, 2011. <<http://docplayer.com.br/712170-Desenvolvimento-de-uma-interface-grfica-para-classificadores-de-imagem.html>>.
- TEIXEIRA, J. P. et al. *Identifica de Caracteres com Rede Neuronal Artificial com Interface Grca*. [S.l.]: Instituto Politico de Bragan 2009. <<http://hdl.handle.net/10198/1868>>.
- WILCOXON, F. Individual comparisons by ranking methods. *Biometrics Bulletin*, v. 1, n. 6, p. 80–83, 1945.

# Implementação

```
obj_Session = obj_JSch.getSession(str_Username, str_Host);  
obj_Session.setPort(int_SSHPort);  
obj_Session.setPassword(str_Password);  
Properties obj_Properties = new Properties();  
obj_Properties.put("StrictHostKeyChecking", "no");  
obj_Session.setConfig(obj_Properties);  
obj_Session.connect();  
java.lang.System.out.println("Connected");
```

```
//CRIA PASTA CAFFE NOVA + .SH EXECUTÁVEL
Channel channel= obj_Session.openChannel("exec");
((ChannelExec)channel).setCommand("cd " + str_Directory +
    "\ncp -r examples/basic/ examples/" + newbase + "/" +
    "\nchmod +x examples/" + newbase + "/*.sh"
);
channel.setInputStream(null);
((ChannelExec)channel).setErrStream(java.lang.System.err);
InputStream in=channel.getInputStream();

channel.connect();
byte[] tmp=new byte[1024];
while(true){
    while(in.available(>0){
        int i=in.read(tmp, 0, 1024);
        if(i<0)break;
        java.lang.System.out.print(new String(tmp, 0, i));
    }
    if(channel.isClosed()){
        java.lang.System.out.println("exit-status: "
            +channel.getExitStatus());
        break;
    }
    try{Thread.sleep(1000);}catch(Exception ee){}
}
channel.disconnect();
```

```

//CRIA E COMPILA IMAGENS CONVERTIDAS
channel= obj_Session.openChannel("exec");
((ChannelExec)channel).setCommand("cd " + str_Directory +
    "\n./examples/" + newbase + "/create_imagenet.sh " + newbase +
    "\nsudo -S -p '' ./build/tools/compute_image_mean -backend leveldb examples/"
    + newbase + "/" + newbase + "_train_leveldb data/" + newbase + "/" + newbase + "_mean.binaryproto"
);
channel.setInputStream(null);
((ChannelExec)channel).setErrStream(java.lang.System.err);
in=channel.getInputStream();
OutputStream out=channel.getOutputStream();

channel.connect();

out.write((str_Password+"\n").getBytes());
out.flush();

tmp=new byte[1024];
while(true){
    while(in.available(>0)){
        int i=in.read(tmp, 0, 1024);
        if(i<0)break;
        java.lang.System.out.print(new String(tmp, 0, i));
    }
    if(channel.isClosed()){
        java.lang.System.out.println("exit-status: "+channel.getExitStatus());
        break;
    }
    try{Thread.sleep(1000);}catch(Exception ee){}
}

channel.disconnect();

```